

## 7. Kenneth

**Program Name:** Kenneth.java

**Input File:** kenneth.dat

Kenneth likes to solve Sudoku puzzles. Sudoku puzzles are not Math puzzles, they are logic puzzles and are a great way to sharpen your problem solving skills. The puzzle consists of a  $9 \times 9$  grid with 9 rows and 9 columns and each cell of the grid will ultimately contain a single digit 1...9. The puzzle starts with some of the cells containing digits and some of the cells empty like the puzzle shown below on the left. The challenge is to fill the remaining cells correctly so that each row and column contains each of the digits 1...9 exactly once, there are no duplicate cells. In addition, each  $3 \times 3$  sub-grid working across and down the puzzle must also contain all of the digits 1...9 with no duplicates. The shading in the following puzzles shows the  $3 \times 3$  sub-grids. The grid on the left is the starting puzzle and the grid on the right is the proposed solution. The following example is a correct solution.

Puzzle

	9						5	
8					4	1		
3	6	2					7	
		1	7	8			3	
	8			6			1	
	4			2	3	6		
	5					7	2	1
		9	2					5
	1						6	

Bold digits are  
start of puzzle

Solution

1	9	4	8	7	2	3	5	6
8	7	5	6	3	4	1	9	2
3	6	2	9	5	1	4	7	8
6	2	1	7	8	9	5	3	4
9	8	3	4	6	5	2	1	7
5	4	7	1	2	3	6	8	9
4	5	6	3	9	8	7	2	1
7	3	9	2	1	6	8	4	5
2	1	8	5	4	7	9	6	3

Row numbers and column numbers are simply 1...9 with row and column 1 in the top left corner of the grid and row and column 9 in the bottom right corner.

The following puzzle and proposed solution contains errors. It has a 2 instead of a 5 in row 2 column 3 and an 8 instead of a 6 in row 7 column 5 causing both row and column errors. Those errors also cause the following sub-grids to contain errors: top-left and bottom-center.

Puzzle

			5		2			
7	8						1	4
	2						5	
		7	1		4	3		
		3	9	5	6	1		
8		1	2		5	4		3
9								5
	5		4		3		7	

Bold digits are  
start of puzzle

Solution

4	1	6	5	7	2	9	3	8
7	8	2	6	3	9	2	1	4
3	2	9	8	4	1	7	5	6
1	6	8	3	2	7	5	4	9
5	9	7	1	8	4	3	6	2
2	4	3	9	5	6	1	8	7
8	7	1	2	8	5	4	9	3
9	3	4	7	1	8	6	2	5
6	5	2	4	9	3	8	7	1

Kenneth is confident in his skills but has asked your UIL programming team for assistance creating a program that confirms whether or not a proposed solution is correct. The program does NOT solve the puzzle.

**Input:** First line will contain a number  $1 \leq T \leq 10$  as the number of test cases. Each test case will be followed by nine lines which are the rows and each row contains exactly 9 digits separated by spaces with all digits being 1...9 and no stray or extra characters.

**Output:** For each test case, the first line of output starts with “GRID #t:” and a space with t as the test case number starting with 1 and that is followed by either “SOLUTION IS CORRECT” or “NOT A SOLUTION”. When the proposed solution is correct, no other details are output. When the proposed solution is not correct, two more lines will be output. The first detail line starts with “>> ROWS WITH ERRORS:” and a space and either a list of row numbers containing errors or “NONE”. The second detail line starts with “>> COLUMNS WITH ERRORS:” and a space and either a list of column numbers containing errors or “NONE”. Display multiple row and column numbers in ascending order separated by single spaces. Each test case result is followed by a row containing 12 equal signs “=====”.

**Sample input:**

3	6 2 1 7 3 9 5 8 4
1 9 4 8 7 2 3 5 6	9 8 3 4 6 5 2 1 7
8 7 5 6 3 4 1 9 2	5 4 7 1 2 3 6 8 9
3 6 2 9 5 1 4 7 8	9 5 6 8 4 3 7 2 1
6 2 1 7 8 9 5 3 4	7 3 9 2 1 6 8 4 5
9 8 3 4 6 5 2 1 7	2 1 8 5 4 7 9 6 3
5 4 7 1 2 3 6 8 9	4 1 6 5 7 2 9 3 8
4 5 6 3 9 8 7 2 1	7 8 2 6 3 9 2 1 4
7 3 9 2 1 6 8 4 5	3 2 9 8 4 1 7 5 6
2 1 8 5 4 7 9 6 3	1 6 8 3 2 7 5 4 9
1 9 4 8 7 2 3 5 6	5 9 7 1 8 4 3 6 2
8 7 5 6 3 4 1 9 2	2 4 3 9 5 6 1 8 7
3 6 2 4 5 1 9 7 8	8 7 1 2 8 5 4 9 3
	9 3 4 7 1 8 6 2 5
	6 5 2 4 9 3 8 7 1

**Sample output:**

```

GRID #1: SOLUTION IS CORRECT
=====
GRID #2: NOT A SOLUTION
>> ROWS WITH ERRORS: NONE
>> COLUMNS WITH ERRORS: 1 4 5 7
=====
GRID #3: NOT A SOLUTION
>> ROWS WITH ERRORS: 2 7
>> COLUMNS WITH ERRORS: 3 5
=====

```